Natural Language Processing

Lab 3: Word Frequency and Part-of-Speech Tagging

Objective

To understand how to visualize word frequency, what part-of-speech tagging is and how to use various libraries to perform part of speech tagging.

Rules

Non-compliance with the following rules will result in serious consequences.

1. Students who are more than 15 minutes late are not allowed to sit in the lab.
2. Monitors will remain off until prompted.
3. Complete silence in the lab.
4. Do not speak in between lectures.
5. For any queries, stay silent and raise your hand.
6. Phones shut off or on silent during lab.
7. Labs will be submitted in hard copy.
8. The previous lab will be submitted in the first 15 minutes of the lab.
9. No late submissions will be accepted.
10. Treat lab tasks as your quiz or lab assignment.
11. No usage of A.I. for lab assignments or lab work.

Word Frequency

Some Natural Language Processing applications require counting the number of times a word occurs and marking its importance on the basis of its count. Having too many words in a text and keeping count is a huge undertaking. Therefore various techniques have been developed to visualize word count.

1. Histogram
2. Word Cloud

Histogram

```
from collections import Counter
import matplotlib.pyplot as plt
```

```
word_freq = Counter(filtered_words)

plt.figure(figsize = (10, 5))
plt.bar(*zip(*word_freq.most_common(10)))
plt.xlabel('Words')
plt.ylabel('Frequency')
plt.title('Top 10 Word Frequencies')
plt.show()
```

Lab Task 1

Plot a histogram of top 20 words from the text story.txt before and after preprocessing.
story.txt: https://shorturl.at/WeE7u

Lab Task 2

Plot a histogram of top 20 words from a blog website (in HTML) before and after preprocessing.

Word Cloud

```
from wordcloud import WordCloud

wordcloud = WordCloud(
    width = 800,
    height = 400,
    background_color = 'white'
).generate_from_frequencies(word_freq)
plt.figure(figsize = (10, 5))
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.show()
```

Lab Task 3

Plot a word cloud of top 20 words from the text story.txt before and after preprocessing.
story.txt: https://shorturl.at/WeE7u

Lab Task 4

Plot a word cloud of top 20 words from a blog website (in HTML) before and after preprocessing.

Part-of-Speech Tagging

Part-of-Speech tagging is a task that involves assigning a grammatical category (such as noun, verb, adjective, etc.) to each word in a sentence. The goal is to understand the syntactic structure of a sentence and identify the grammatical roles of individual words. POS tagging provides essential information for various NLP applications, including text analysis, machine translation, and information retrieval.

Lab Task 5

Take 3 sentences and manually tag each word after preprocessing and print them as tuples `(word, tag)`.

NTLK

```
import nltk
from nltk.tokenize import word_tokenize
from nltk import pos_tag

nltk.download('punkt_tab')
nltk.download('averaged_perceptron_tagger_eng')

sentence = "The quick brown fox jumps over the lazy dog."
tokens = word_tokenize(sentence)
pos_tags = pos_tag(tokens)
```

Lab Task 6

Take 3 sentences and tag each word after preprocessing with NLTK and print them as tuples `(word, tag)`.

SpaCy

SpaCy is a library for advanced Natural Language Processing in Python and Cython. It's built on the very latest research, and was designed from day one to be used in real products.

```
import spacy

nlp = spacy.load("en_core_web_sm")

doc = nlp("The quick brown fox jumps over the lazy dog.")
for token in doc:
    print(f"{token.text}: {token.pos_}")
```

Lab Task 7

Take 3 sentences and tag each word after preprocessing with spaCy and print them as tuples `(word, tag)`.

Lab Task 8

Perform POS tagging of story.txt after preprocessing with both NLTK and spaCy and measure the time difference between them.
story.txt: https://shorturl.at/WeE7u